

A Novel Approach for Automation of Precision Calibration Process

Na Li¹, Qin Liu², Lili Wang³, Xiaoyi Liu⁴
School of Software Engineering, Tongji University
Shanghai 200092, China

lina_sh@yahoo.cn¹, qin.liu@mail.tongji.edu.cn², sdxtlily@gmail.com³, andrea.liu.cn@gmail.com⁴

Abstract—Precision calibration of industrial sensors and instruments are usually performed manually in the calibration laboratory. In this paper, a computer controlled automated solution with a software prototype controlling the whole calibration process is proposed. This solution minimizes the human involvement and therefore reduces the chances of errors. Data acquiring, processing and calculation are simplified by instruments communication through General Purpose Interface Bus (GPIB). The automation has been practically conducted by a laboratory and calibration efficiency has been improved.

Keywords—automation, GPIB, calibration process, instrument communication, camera control

I. INTRODUCTION

The validity of measurement made by industrial sensors and instruments is ensured by regular calibration, which provides traceability against to a national standard. In other words, a field sensor (e.g. hygrometer, thermometer, etc.) must be calibrated against a reference system which is composed of high quality, reliable reference instruments.

Although many companies and organizations are using various calibration methods and processes to ensure high accuracy of instrument calibration, the common calibration process is operated manually which can be generalized as follows:

1. The industrial sensor or instrument to be calibrated is placed inside the calibration chamber with its display outside the chamber;
2. Connecting reference instruments with calibration chamber;
3. According to certain required standards, configuring measurement environment in the calibration chamber (e.g. temperature, dew point, air pressure, humidity, etc.) through those control buttons outside the chamber;
4. Waiting until the measurement environment conditions reach the stable state and then obtaining measurement results of industrial sensor and reference instruments;
5. Setting different measurement environment conditions according to certain requirements, and then getting all the measurement results;
6. After data acquisition, the validity and accuracy of the industrial sensor is analyzed and calculated, based on

specific formulas and national standards.

Even with the high quality reference instruments, which can be considered reliable, the drawbacks of calibration process itself are obvious. First, it is a huge waste of time and human resource since the overall calibration process could last days even over weeks, and an operator has to stay in the laboratory all the time repeating calibration and measurement processes. Second, no matter how accurate the reference instruments are, measurement results are read and calculations are done manually. The inaccuracy and mistakes are unavoidable.

In order to solve those problems, many efforts have been made by research laboratories and institutes around the world. Jovan Bojkovski, Janko Drnovsek et al. discuss the general issues of precision calibration automation and several guidelines for corresponding software design in [1]. Later in [2], they present an automated system for evaluation of climatic chambers in the laboratory as well as on site, although the system which needs human involvement is not completely automated and the cost of the hardware equipment is not economic for small laboratories. Lately, in [3], V. Batagelj et al. state the concepts and design approaches of the measurement-automation software. However, units under testing without a standard GPIB interface are not considered in their solution. Compared to those positive attempts, in [4], H.K. Chan expatiates an unsuccessful attempt to automate calibration processes for electronic measuring devices in an instrumentation laboratory and suggestions on how to address the human problem that encounters.

Unlike those approaches mentioned above, in this paper, we propose a module-based solution to automatize the calibration process. A software prototype controlling the overall process is developed and has been implemented in a laboratory. It highly improves the previous manual process and saves the cost of time and human resource. Also, this automated approach enables a full compliance with a standard interface of the reference instruments and diminishes human errors.

The rest of the paper is organized as follows. In section II, common problems in manual calibration process are described. Section III proposes an automatic approach with a low cost and high efficiency. In section IV, a case study of proposed approach in a typical precision calibration laboratory is presented. The cost and efficiency is compared between the

manual and automatic process. The conclusions and future work are stated in section V.

II. PROBLEM DEFINITION

A. Problem Description

Usually the precision calibration laboratory buys reference measuring instruments and programs for those instruments from various producers. Because those programs are not compatible in most cases, the laboratory has to control the whole calibration process manually in order to fulfill their specific requirements. For example, different measurement environment should be configured manually, measurement results are acquired by person or by specific programs, and finally calculations and analysis is performed using another program or even manually. Obviously, automation of the calibration process is necessary and general software which is applicable for most calibrations has to be designed.

B. Key Issues

In order to design such a program, attention should be paid to key issues in the calibration process.

1. Almost all the reference measuring instruments are programmable instruments, which have the IEEE Standard interface-GPIB. The GPIB was specifically designed to connect computers and laboratory instruments so that data and control information can pass between them [7]. Instead of reading from instrument display by the operator, acquiring measurement results through GPIB is more accurate and convenient.
2. Compared to reference instruments, industrial measuring unit under calibration hardly has a GPIB interface. Furthermore, they have different sizes, different shapes and even different ways to display measurement results. A general solution to get their measurement results should be found.
3. It takes some time for the environment in calibration chamber to get into a stable state after environment parameters are configured. In previous manual calibration process, the operator can decide whether the stable state has been reached through watching measurement results of reference instruments. Accordingly, this issue also needs to be solved in the automatic process.
4. The way how measurement results are acquired should be decided since fluctuation happens sometimes. If results are read with a certain frequency like every five minutes, inaccurate results (i.e. peak or nadir of the fluctuation) could be got. So an appropriate method of reading measuring results should be defined.

C. Literature Review

Automation of measuring instruments calibration has been developing since the last century, thus various methods and solutions have been tried again and again. In 1965, Hewlett-Packard designed the Hewlett-Packard Interface Bus (HP-IB) to connect their line of programmable instruments to their

computers. It was later standardized as IEEE Standard 488-1975, and ANSI/IEEE 488.2-1987 strengthened the original standard by defining precisely how controllers and instruments communicate. Today, the name General Purpose Interface Bus (GPIB) is more widely used than HP-IB. Standard Commands for Programmable Instruments (SCPI) took the command structures defined in IEEE 488.2 and created a single, comprehensive programming command set that is used with any SCPI instrument [8].

III. PROPOSED APPROACH

A. Hardware Architecture

Based on the discussion in section II, hardware architecture of the automatic calibration process can be designed.

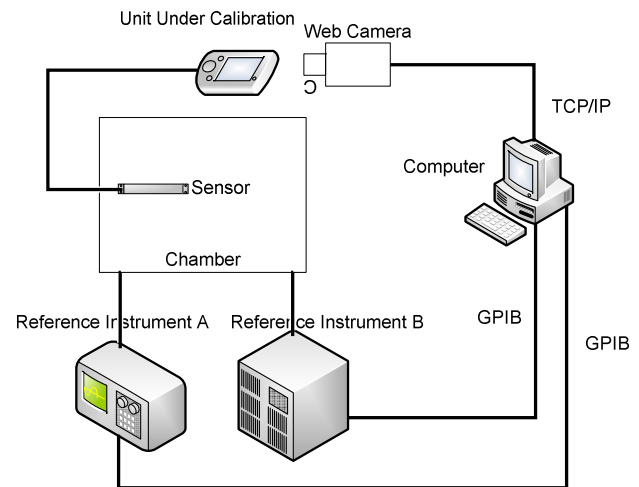


Figure 1. Hardware Architecture

Instead of controlling the whole process by human, a computer is needed to run the controlling software and connect with the reference measuring instruments. Almost all the reference measuring instruments are programmable instruments, which have the IEEE Standard interface-GPIB. We can take advantage of the interface by using SCPI to control the instruments and acquire raw data from them. Controlling reference instruments through GPIB interface minimizes the human involvement and therefore reduces the possibility of errors.

Compared to reference instruments, industrial measuring unit under calibration hardly has a GPIB interface. Considering the variety of industrial measuring units under calibration, it is impossible to acquire the measuring results with the same interface. However, all of them have got a display to show their measuring results. As a result, taking photos of the display with a web camera automatically would be a choice. In this way, the controlling software takes the responsibility of web camera control so that the operator doesn't have to wait in the laboratory while measuring. When all the measurements are finished, the measuring results are recorded by the operator according to the photos taken.

After the analysis above, we can draw a conclusion that the hardware architecture in the calibration process can be designed as in Figure 1. The sensor of unit under calibration is placed inside the chamber and the web camera focuses on the display of the unit. The web camera is connected to and controlled by the computer through TCP/IP. Reference measuring instruments like hygrometer and barometer are connected to the chamber while GPIB links these instruments to the USB interface of the computer.

B. Software Module Design

After the discussion about hardware architecture, the module-based software system which runs on the computer and controls the overall calibration process can be designed.

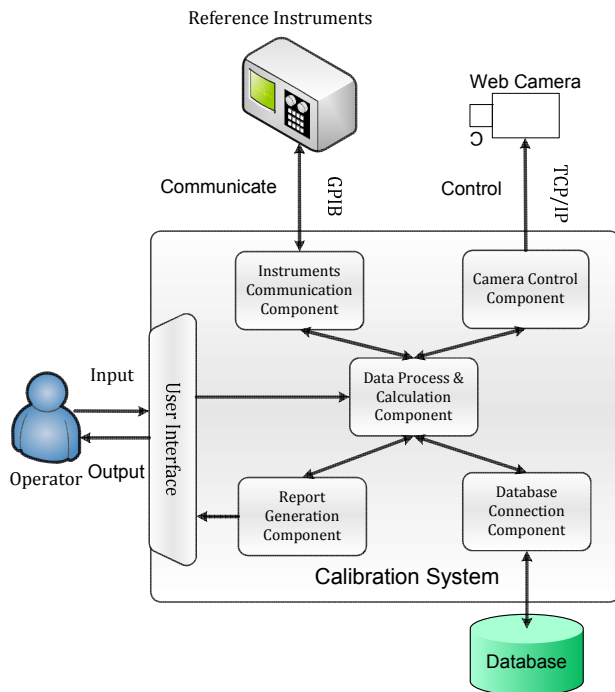


Figure 2. Software Architecture of Calibration System

As shown in Figure 2, the calibration system is composed of instruments communication component, camera control component, data process & calculation component, report generation component, database connection component and user interface. Instruments communication component sends commands and receives data from reference instruments through GPIB, camera control component captures the video stream and takes photos of the web camera, data process & calculation component invokes and controls all the other components.

C. Automation of Calibration Process

1) Automatic Calibration Process

Based on the hardware and software architecture of the calibration system, the automation of the overall process can be described as follows.

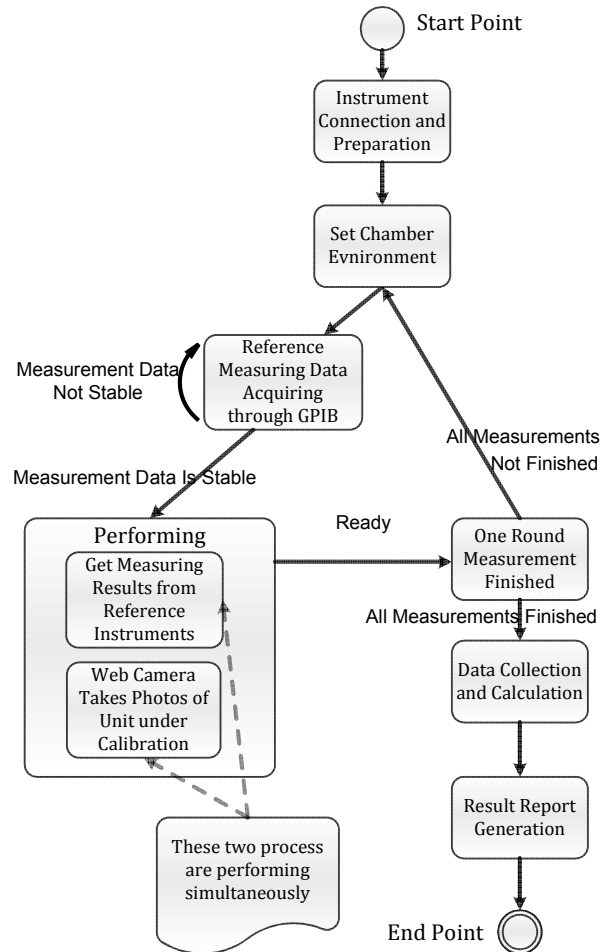


Figure 3. Automated Calibration Process

As shown in Figure 3, the automated calibration process is performed in seven steps.

1. The operator puts the sensor of the unit to be calibrated into the chamber, connects all the instruments including the web camera, making sure that the camera focuses on the display of the unit.
2. Then the operator configures the testing environment through the control buttons on the chamber.
3. After all the preparations are ready, the software starts acquiring measuring data from reference measuring instruments through GPIB. Since it takes some time for the environment to become stable, this period will not end up until the measurement data is stable.
4. When it's stable, the software begins to read the measuring results from the reference instruments through GPIB, and at the same time, web camera takes photos of the unit display under the control of the software. When data recording and storage is done, one round of measurement has finished.

5. Usually the unit should be tested under several different environments, so step 2, 3 and 4 should be executed several times until all the measurements are taken.
6. After finishing all the measurements, the software collects the resultant data of all the measurements and displays the corresponding resultant photos. When the operator inputs the measuring results of the unit under calibration according to the photos, the software performs the raw data processing and calculations.
7. Finally a result report is generated by the software.

2) Intervals of Data Acquisition

During the calibration, usually several different groups of environment parameters (temperature, humidity, air pressure, etc.) should be configured so that the industrial unit can be tested under different environments. For one certain environment, in order to ensure accuracy, our software is supposed to acquire the measuring results several times and then get an average one. So the intervals between different acquisitions should be considered. Because there are fluctuations in the measuring results during one measurement, inaccurate data (i.e. peak or nadir of the fluctuation) might be obtained if acquiring data with a fixed interval. In view of this problem, the intervals should be random so that the average result will be more close to the actual one. Suppose T_n stands for the time when the computer is acquiring data for the n th time, and R_i stands for a random time period then the equation can be shown as in (1).

$$T_{n+1} = T_n + R_i. \quad (1)$$

D. Instruments Communication

During the overall calibration process, instruments communication through GPIB is of great importance. As mentioned before, IEEE 488.2 has defined SCPI, which can be used to control and communicate with the reference measuring instruments. National Instruments Corporation provides a class library called NI-488.2, which is included when the NI-488.2 driver is installed. The NI-488.2 class library includes a set of classes for communicating with GPIB instruments, controlling GPIB devices, and acquiring GPIB status information [9]. This library can be used to design code that communicates with and controls instruments on a GPIB interface. C# code of instruments communication using this library will be discussed next.

1) Device Initialization & Read Operation

In order to communicate with an instrument, a construct function should be invoked first so that the certain instrument can be identified by its board number, primary address and secondary address. After device initialization, read operation can be executed on the specified instrument to acquire measuring data. Sometimes the instrument returns nothing or an empty value for some reason after the computer sends a read command. It should be handled in case that an exception happens later on. The C# code is as follows.

```
//C# code for device initialization
Device myDevice = new Device ( (int) boardNumber,
                               (byte) primaryAddress,
                               (byte) secondaryAddress );
```

```
myDevice.Write( "READ?");
string str = myDevice.ReadString();
while (str == " _EMPTY_ ")
{
    myDevice.Write( "READ?");
    str = myDevice.ReadString();
}
```

2) Timeout Control

In some cases, response delay happens after the instrument receives the command sent by the computer through GPIB. If the time delay is not handled well, it may lead to process failure or program crash. A solution to that is to define a time limit for the response delay, once the response time of IO operation exceeds the time limit, an exception will be caught and handled. The C# code is as follows.

```
//C# code for handling timeout of instrument response
TimeoutValue myTimeOut = TimeoutValue.T3s;
try
{
    myDevice.IOTimeout = this.myTimeOut;
}
catch (NationalInstruments.NI4882.GpibException ge)
{
    System.Windows.Forms.MessageBox.Show("Timeout! Could not
    connect to the instrument!");
}
```

E. Camera Control

Another part of the automation is that the web camera is supposed to take photos of the unit display under control of the software. A web camera has its own IP address when it is connected to the network. It captures and transmits live images directly over an IP network, enabling authorized users to locally or remotely view, store, and manage video over standard IP-based network infrastructure [10].

In our approach, a local area network (LAN) will be formed when the web camera is connected to the computer with the network wire. The software identifies the web camera by its IP address and controls the camera with the class library included in the camera driver. Since the sizes and shapes of the units to be calibrated are different from one another, the operator should make sure whether the camera focuses on the display of the unit. In our software, the video of the web camera is shown on user interface so that the operator can see directly from the video what the web camera is focusing on and whether the picture is clear enough. The operator can adjust the focus of the camera until the picture is correct and clear.

Our software not only captures the video stream of the web camera and plays it on user interface, but also controls the camera to take photos of the testing unit with certain intervals mentioned before. Since the photos are stored for further use, the operator doesn't have to stay in the laboratory waiting and recording during the long time of measuring.

IV. CASE STUDY

The automation has been practically conducted by a precision calibration laboratory in Denmark, and the user interface of the software prototype is shown as follows.

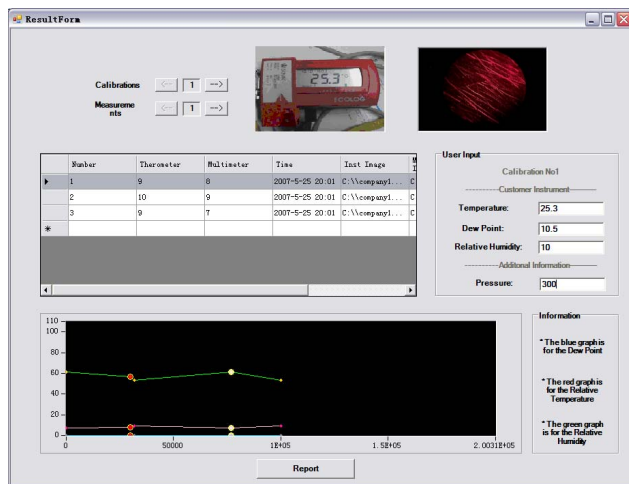


Figure 4. User Interface of Calibration Software

TABLE I. COMPARISON BETWEEN MANUAL AND AUTOMATIC PROCESS

	Manual Process	Automatic Process
Human Resource Cost	Overall process is carried out by the operator	Reduced by 60%
Time Cost	Data acquiring, waiting and calculation performed by human	Software controlled, Reduced by 20%
Complexity of Work	Operator stays in the laboratory all the time, reading, waiting and calculating	Operator clicks some buttons, Reduced by 80%
Human Errors	Reading, typing and calculating errors	Reduced by 50%
Visualization	Measuring data is recorded by human.	Measuring data is illustrated clearly with graph.
Traceability and Maintainability	Calibration results and reports are kept by paper work.	Calibration results are stored in data base and reports are kept in electric files.

As shown in Figure 4, there are two photos shown on the top part of the window. The left one is the photo of testing unit and the right one is the photo of chamber's eyepiece. The graph on the bottom of the window, which is drawn with Measurement Studio [9], illustrates the measuring results of reference instruments. The operator can see clearly whether the measuring data is stable or not and how the data fluctuates. After the inputs of report requirements, a result report will be generated automatically.

Evaluation and statistics of the precision calibration laboratory show that, the cost for human resource decreases and the possibility of human errors reduces after the laboratory puts the automatic solution into practice. A comparison between the automatic solution and previous manual

calibration process are shown in Table I. The statistics are given by the laboratory.

V. CONCLUSIONS AND FUTURE WORK

Precision calibration of industrial sensors and instruments are usually carried out in the laboratory manually, which brings lots of cost of human resource and especially increases the possibility of human errors. In this paper, we propose an automatic solution with low cost and high efficiency. The most important contribution is the software controlling the overall calibration process, which is applicable for most calibrations. Our approach reduces human involvement, simplifies the calibration process and strengthens traceability of calibration. The automatic solution and the software have been put into practice in a precision calibration laboratory. According to the feedback, high efficiency, good functionality and great reliability have been shown compared to the previous manual process.

Though, there are still some issues we should improve. For one thing, every time the testing environment needs to be changed, the operator has to configure the environment parameters by operating the control buttons on the chamber. If the configuration can be controlled by software, then the operator can be totally released from the calibration process. For another, after finishing all the measurements, the operator has to record the measuring results of the testing unit all by himself/herself according to the photos taken by the web camera. We are investigating the possibility of recognizing the measuring data in the photos, in order to minimize human involvement in the calibration process.

REFERENCES

- [1] Bojkovski J, Drnovsek J, Pusnik I, Tasic T, "Automation of a precision temperature calibration laboratory", *IEEE Transaction on Instrument and Measurement*, vol. 3, pp. 1883-1887, Jun 2000.
- [2] Jovan Bojkovski, Domen Hudoklin, Janko Drnovsek et al., "Custom made automated system for determination of humidity and temperature gradients in climatic chambers", *Instrumentation and Measurement Technology Conference*, vol. 1, pp. 2-5, May 2003.
- [3] V. Batagelj, Jovan. Bojkovski, Janko. Drnovsek, "Software integration in national measurement-standards laboratories", *IET Science, Measurement & Technology*, vol. 2, pp. 100-106, Mar 2008.
- [4] H.K. Chan, "Resistance-something behind automated calibration", *IEEE Instrumentation & Measurement Magazine*, vol. 8, pp. 60-62, Dec 2005.
- [5] R. Lapuh, "Automation of a small calibration laboratory", *Abstracts of the 1999 NCSL Workshop and Symp.*, pp.42, Jul 1999.
- [6] Serge Demidenko, Sayne Moorhead, "Electronic test technology curriculum revisiting", *the Third IEEE International Workshop on Electronic Design, Test and Applications*, pp. 5-9, Jan 2006.
- [7] Tech Soft GmbH, "IEEE-488 Tutorial", 1997-2005.
- [8] National Instruments, "GPIB Tutorial", 1998-2008.
- [9] National Instruments, "Measurement Studio User Manual", 2003-2006.
- [10] Axis Communications, "AXIS Media Control User Manual", 2007-2008.
- [11] Electronics Group, "GPIB Programming Tutorial", Jan 2000.
- [12] Scott J. Richardson, "Automated Temperature and Relative Humidity Calibrations for the Oklahoma Mesonet", *Journal of Atmospheric and Oceanic Technology*, vol. 12, pp. 951-959, Aug 1995.